

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-353156

(43)Date of publication of application : 24.12.1999

(51)Int.Cl.

G06F 7/50

(21)Application number : 10-160204

(71)Applicant : HITACHI LTD

(22)Date of filing : 09.06.1998

(72)Inventor : EBATA ATSUSHI
YAMAMOTO MICHITAKA
KATO TAKESHI

(54) CARRY SIGNAL GENERATING CIRCUIT

(57)Abstract:

PROBLEM TO BE SOLVED: To make an adder fast by shortening the total carry signal generation time including precedent and trailing stages as to a precedent-stage circuit and a trailing-stage circuit which constitute the carry signal generating circuit by precedently processing part of logical operation of the trailing-stage circuit by the precedent-stage circuit.

SOLUTION: When carry signals for 4-bit binary data $a_0a_1a_2a_3$ and binary data $b_0b_1b_2b_3$ are generated, a signal p_0 represented as $p_0=a_0+b_0$, a signal H_i , $i+1$ represented as H_i , $i+1=a_i.b_i+a_{i+1}$ ($i=0, 2$), and a signal $I_{1,2}$ represented as $I_{1,2}=(a_1+b_1).(a_2+b_2)$ are generated. Their signal processings can be carried out in parallel. Then those signals p_0 , H_i , $i+1$, and $I_{1,2}$ are used to generate a 4-bit carry signal C_{4bit} represented as $C_{4bit}=p_0.(H_0, 1+I_{1,2}+H_2, 3)$. Consequently, the number of NMOS transistors which are stacked and connected longitudinally is decreased to shorten the circuit operation time.

LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-353156

(43) 公開日 平成11年(1999)12月24日

(51) Int.Cl.⁶

G 0 6 F 7/50

識別記号

F I

G 0 6 F 7/50

G

審査請求 未請求 請求項の数 4 O L (全 9 頁)

(21) 出願番号 特願平10-160204

(22) 出願日 平成10年(1998)6月9日

(71) 出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72) 発明者 江端 淳

東京都国分寺市東恋ヶ窪一丁目280番地

株式会社日立製作所中央研究所内

(72) 発明者 山本 通敬

神奈川県秦野市堀山下1番地 株式会社日

立製作所汎用コンピュータ事業部内

(72) 発明者 加藤 猛

東京都国分寺市東恋ヶ窪一丁目280番地

株式会社日立製作所中央研究所内

(74) 代理人 弁理士 小川 勝男

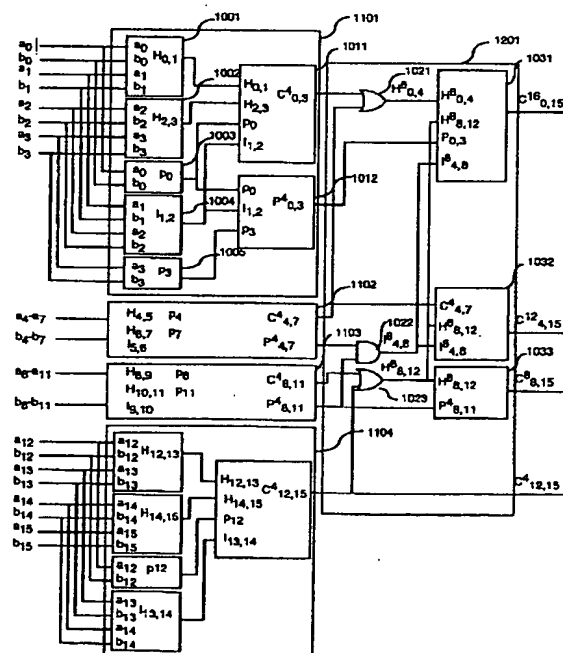
(54) 【発明の名称】 桁上げ信号生成回路

(57) 【要約】

【課題】 加算器の桁上げ信号を生成する論理回路の高速化をはかる。

【解決手段】 4ビットの2進データ a_i と b_i の桁上げ信号生成回路を例題として、前段回路群により信号 $p_0 = a_0 + b_0$ と信号 $H_{i,i+1} = a_i \cdot b_i + a_{i+1} \cdot b_{i+1}$ ($i = 0, 2$) と信号 $I_{1,2} = (a_1 + b_1) \cdot (a_2 + b_2)$ を生成し、これらから後段回路により桁上げ信号 $C^{4bit} = p_0 \cdot (H_{0,1} + I_{1,2} \cdot H_{2,3})$ を生成する。

図 1 2



【特許請求の範囲】

【請求項1】4ビットの2進データ a_0, a_1, a_2, a_3 と2進データ b_0, b_1, b_2, b_3 との加算器の桁上げ信号生成回路において、ブール代数式 $a_i + b_i$ で表される信号 p_i を生成する回路と、ブール代数式 $a_i \cdot b_i + a_{i+1} \cdot b_{i+1}$ ($i=0, 2$)で表される信号 H_{i+1} を生成する回路と、ブール代数式 $(a_1 + b_1) \cdot (a_2 + b_2)$ で表される信号 $I_{1,2}$ を生成する回路と、ブール代数式 $p_0 \cdot (H_{0,1} + I_{1,2} \cdot H_{2,3})$ で表される4ビットの桁上げ信号を生成する回路とを有することを特徴とする桁上げ信号生成回路。

【請求項2】4ビットの2進データ a_0, a_1, a_2, a_3 と2進データ b_0, b_1, b_2, b_3 との加算器の桁上げ信号生成回路において、前記データの反転信号 a_0B, b_1B, b_2B, b_3B と b_0B, b_1B, b_2B, b_3B とにより、ブール代数式 $a_0B \cdot b_0B$ で表される信号 p_0B を生成する回路と、ブール代数式 $(a_iB + b_iB) \cdot (a_{i+1}B + b_{i+1}B)$ ($i=0, 2$)で表される信号 $H_{i+1}B$ を生成する回路と、ブール代数式 $a_1B \cdot b_1B + a_2B \cdot b_2B$ で表される信号 $I_{1,2}B$ を生成する回路と、ブール代数式 $p_0B + H_{0,1}B \cdot (I_{1,2}B + H_{2,3}B)$ で表される4ビットの桁上げ信号の反転信号を生成する回路とを有することを特徴とする桁上げ信号生成回路。

【請求項3】請求項1または請求項2記載の桁上げ信号生成回路において、前記ブール代数式で表わされる回路をCMOSドミノ論理回路により構成したことを特徴とする桁上げ信号生成回路。

【請求項4】 n ビット (n は4以上の整数)の2進データ $a_0, a_1, a_2, a_3, \dots, a_n$ と2進データ $b_0, b_1, b_2, b_3, \dots, b_n$ の加算器の桁上げ信号生成回路において、上記桁上げ信号生成回路を請求項1または請求項2記載の4ビット桁上げ信号生成回路に基づいて構成したことを特徴とする桁上げ信号生成回路。

【発明の詳細な説明】

【0001】

*

【0006】

$$g_j = a_j \cdot b_j \quad (j=0, 1, 2, 3) \quad \dots (1)$$

$$p_j = a_j + b_j \quad (j=0, 1, 2, 3) \quad \dots (2)$$

$$C^{4bit} = g_0 + p_0 g_1 + p_0 p_1 g_2 + p_0 p_1 p_2 g_3 \\ = g_0 + p_0 \cdot (g_1 + p_1 \cdot (g_2 + p_2 \cdot g_3)) \quad \dots (3)$$

従来技術はこれらの信号 g_j, p_j, C^{4bit} をドミノ論理回路を用いて生成することにより高速化を図っている。

【0007】図9は生成信号 g_j を生成するドミノ回路、図10は伝搬信号 p_j を生成するドミノ回路、図11は桁上げ信号 C^{4bit} を生成するドミノ回路を示す。

【0008】図9において、301はクロック信号CKが入力されるプリチャージ用PMOSTランジスタ、304はCMOSインバータである。NMOSTランジスタ302、303にはデータ a_j と b_j が入力されている。クロック信号CKが論理値1のとき、データ a_j と b_j の論理積である生成信号 g_j がインバータ304を介して出

【発明の属する技術分野】本発明は、データ信号加算器の桁上げ先を行なう論理回路に係り、特に桁上げ信号生成の高速化に好適なCMOS論理回路に関する。

【0002】

【従来の技術】マイクロプロセッサやデジタル信号処理プロセッサ等の情報処理装置に用いられる論理回路の中で、加算器は最も基本的な構成要素の一つである。多くの場合に加算器が情報処理装置の動作速度を決めているため、その高速化が強く求められている。

【0003】多ビット加算器の信号処理速度は桁上げ信号を生成する時間に大きく依存しており、桁上げ信号生成回路が加算器全体の処理速度を決める重要な支配要因となっている。このため、従来から桁上げ信号生成回路を高速化する手法が検討されてきた。

【0004】従来の桁上げ信号生成回路の高速化技術としては、エヌ・エッチ・イー・ウエステ、ケー・エシュライアン著、プリンシプルズ・オブ・シーモス・ブイエールエスアイ・デザイン・—・ア・システムズ・パースペクティブ、第2版、アディソン・ウェズレー出版社、第526頁〜第531頁、1993年(N.H.E. Weste and K. Eshraghian, Principles of CMOS VLSI Design—A Systems Perspective, second edition, Addison-Wesley, pp. 526-531, 1993)に記載のようなダイナミック回路(ドミノ回路)を用いたCMOS論理回路がある。ドミノ論理回路は、高速回路技術として一般的に広く知られている。

【0005】上記従来技術では、4ビットの2進データ a_0, a_1, a_2, a_3 と b_0, b_1, b_2, b_3 を加算する場合、まず、下記のブール代数式(1)と式(2)に示すように上位から j ビット目のデータ a_j と b_j の論理積から生成信号 g_j を生成し、 a_j と b_j の論理和から伝搬信号 p_j を生成する。次に、式(3)に示すようにこれらの生成信号 g_j と伝搬信号 p_j を用いて4ビットの桁上げ信号 C^{4bit} を生成している。

【0006】

力される。例えば、 a_j, b_j の両方が論理値1に成った場合には、トランジスタ301、302が導通し、インバータ304の入力が論理値0になり、桁上げ生成信号 g_j が論理値1になる。

【0009】図10の伝搬信号生成回路では、311がクロック信号CKが入力されるプリチャージ用PMOSTランジスタ、314がCMOSインバータである。NMOSTランジスタ312、313にはデータ a_j と b_j が入力されている。クロック信号CKが論理値1のとき、データ a_j と b_j の論理和である伝搬信号 p_j がインバータ314を介して出力される。

【0010】図11の4ビットの桁上げ信号生成回路では、図9と図10の回路によって生成された生成信号 $g_0 \sim g_3$ と伝搬信号 $p_0 \sim p_3$ とが、それぞれに対応するトランジスタ323～328に入力されている。プリチャージ用PMOSTランジスタ321に入力されるクロック信号CKが論理値1のとき、前記式(3)で表わされた論理演算が実行され、その結果である桁上げ信号 C^{4bit} がインバータ329から出力される。

【0011】

【発明が解決しようとする課題】上記従来技術では、データ a_j と b_j から4ビットの桁上げ信号 C^{4bit} が生成されるまでの信号処理時間は、図9のNMOSTランジスタ302、303とインバータ304の動作時間と、図11のNMOSTランジスタ322、324、326、328とインバータ329の動作時間で決まる。すなわち、信号処理速度が、縦積み接続されたNMOSTランジスタ2+4個とインバータ2個の動作時間によ *

$$p_0 = a_0 + b_0 \quad \dots (4)$$

$$H_{i,i+1} = a_i \cdot b_i + a_{i+1} \cdot b_{i+1} \quad (i = 0, 2) \quad \dots (5)$$

$$I_{1,2} = (a_1 + b_1) \cdot (a_2 + b_2) \quad \dots (6)$$

$$C^{4bit} = p_0 \cdot (H_{0,1} + I_{1,2} \cdot H_{2,3}) \quad \dots (7)$$

式(4)～式(7)をドミノ論理回路で構成した場合、式(4)は縦積みNMOSTランジスタ1個とインバータ1個の回路、式(5)と式(6)は縦積みトランジスタ2個とインバータ1個の回路、式(7)は縦積みトランジスタ3個とインバータ1個の回路で構成される。式(4)～式(6)の信号処理は並行できるので、データ a_j と b_j から桁上げ信号 C^{4bit} を生成するまでの信号処理時間は、式(5)または式(6)の処理時間と式(7)の処理時間とに依存する。したがって、桁上げ信号 C^{4bit} の生成時間は縦積みNMOSTランジスタ2+3個とインバータ2個の動作時間で決まり、従来技術に※

$$p_0 B = a_0 B \cdot b_0 B \quad \dots (8)$$

$$H_{i,i+1} B = (a_i B + b_i B) \cdot (a_{i+1} B + b_{i+1} B) \quad (i = 0, 2) \quad \dots (9)$$

$$I_{1,2} B = a_1 B \cdot b_1 B + a_2 B \cdot b_2 B \quad \dots (10)$$

$$C^{4bit} B = p_0 B + H_{0,1} B \cdot (I_{1,2} B + H_{2,3} B) \quad \dots (11)$$

式(8)～式(11)をドミノ論理回路で構成した場合、式(8)～式(11)はいずれも縦積みNMOSTランジスタ数が2個でインバータ1個の回路になる。したがって、データ a_j と b_j から桁上げ信号の反転信号 $C^{4bit} B$ を生成するまでの信号処理時間は、縦積みトランジスタ2+2個とインバータ2個の動作時間で決まり、上記第1の手段に比べて縦積みトランジスタ数をさらに1個削減できる。

【0017】

【発明の実施の形態】まず、本発明による4ビット桁上げ信号生成回路の第1の実施例を図1～図4により説明する。図1～図4は、それぞれ式(4)～式(7)に対応する論理演算を行うドミノ回路を示す。

【0018】図1は、4ビットの2進データの上位から 50

*って支配されている。

【0012】本発明の主な目的は、桁上げ信号生成回路をさらに高速化することにある。論理的な工夫によって、従来技術よりも縦積み接続されたNMOSTランジスタの個数を削減し、回路動作時間を短縮することが狙いである。

【0013】

【課題を解決するための手段】上記目的を達成するため、本発明の第1の手段では、4ビットの2進データ $a_0 a_1 a_2 a_3$ と2進データ $b_0 b_1 b_2 b_3$ の桁上げ信号を生成する場合を例にとると、まず式(4)で表わされる信号 p_0 と式(5)で表される信号 $H_{i,i+1}$ と式(6)で表わされる信号 $I_{1,2}$ を生成する。次に、これらの信号 p_0 、 $H_{i,i+1}$ 、 $I_{1,2}$ を用いて式(7)で表わされる4ビットの桁上げ信号 C^{4bit} を生成する。

【0014】

※比べて縦積みトランジスタ数を1個削減できる。

【0015】本発明の第2の手段では、データ $a_0 a_1 a_2 a_3$ と $b_0 b_1 b_2 b_3$ の反転信号 $a_0 B a_1 B a_2 B a_3 B$ と $b_0 B b_1 B b_2 B b_3 B$ を利用して、まず式(8)で表わされる信号 $p_0 B$ と式(9)で表される信号 $H_{i,i+1} B$ と式(10)で表わされる信号 $I_{1,2} B$ を生成する。次に、これらの信号 $p_0 B$ 、 $H_{i,i+1} B$ 、 $I_{1,2} B$ を用いて式(11)で表わされる4ビットの桁上げ信号の反転信号 $C^{4bit} B$ を生成する。

【0016】

0ビット目のデータ a_0 と b_0 の論理和すなわち伝搬信号 p_0 を生成する回路である。この回路は、クロック信号CKが入力されるPMOSTランジスタ101と、データ a_0 が入力されるNMOSTランジスタ102と、データ b_0 が入力されるNMOSTランジスタ103と、CMOSインバータ104から構成されている。PMOSTランジスタ101の上部ノードは電源に、NMOSTランジスタ102と103の下部ノードはグランドに接続されている。

【0019】クロック信号CKが論理値0の時、ドミノ回路はプリチャージ期間にある。この期間には a_0 と b_0 の論理演算が行われず、インバータ104の出力が論理値0になる。CKが論理値1になると、ドミノ回路がエバリュエーション期間に入り、式(4)に対応する a

5

と b_0 の論理和の演算が行われる。データ a_0 , b_0 のどちらか一方が論理値1になると、インバータ104の入力信号が論理値0になり、出力信号すなわち伝搬信号 p_0 が論理値1になる。この回路の信号処理時間は、トランジスタ102または103によりダイナミックノード(PMOSとNMOSの間のノード)のチャージが引き抜かれる時間と、ダイナミックノードの信号遷移がインバータ104を介して出力される時間との和になる。

【0020】図2は、4ビットの上位から i ビット目($i=0$ または2)のデータ a_i と b_i の論理積すなわち生成信号 g_i を生成し、 $i+1$ ビット目のデータ a_{i+1} と b_{i+1} との論理積すなわち生成信号 g_{i+1} を生成し、 g_i と g_{i+1} の論理和信号 H_{i+1} を生成する回路である。プリチャージ用PMOSトランジスタ111と、データ a_i , b_i , a_{i+1} , b_{i+1} が入力されるNMOSトランジスタ112, 113, 114, 115と、インバータ116から構成されている。

【0021】クロック信号CKが論理値1になり、ドミノ回路がエバリュエーション期間に入ると、式(5)に対応する生成信号 g_i と生成信号 g_{i+1} の論理和演算が行わ

れる。この回路の動作速度は、縦に2個直列接続されたNMOSトランジスタ112と113の対または114と115の対がダイナミックノードをディスチャージする時間と、インバータ116が切り替わる時間で決ま

る。【0022】図3は、4ビットの上位から1ビット目のデータ a_1 と b_1 の論理和すなわち伝搬信号 p_1 を生成し、2ビット目のデータ a_2 と b_2 の論理和すなわち伝搬信号 p_2 を生成し、 p_1 と p_2 の論理積信号 $I_{1,2}$ を生成する回路である。プリチャージ用PMOSトランジスタ121と、データ a_1 , b_1 , a_2 , b_2 が入力されるNMOSトランジスタ122, 123, 124, 125と、インバータ126から構成されている。

【0023】この回路がエバリュエーション期間に入ると、式(6)に対応する論理積信号 $I_{1,2}$ が出力される。プリチャージ期間には信号 $I_{1,2}$ が論理値0になる。エバリュエーション期間にNMOSトランジスタ122と123の少なくとも一方と、124と125の少なくとも一方とがオン状態になった場合に信号 $I_{1,2}$ の論理値が0から1に切り替わる。信号処理時間は、縦2個同時に導通したNMOSトランジスタがダイナミックノードをディスチャージする時間と、インバータ126が動作する時間との和で決まる。

【0024】図4は、図1から出力された伝搬信号 p_0 と図2から出力された信号 $H_{0,1}$, $H_{2,3}$ と図3から出力された信号 $I_{1,2}$ を受けて、式(7)に対応する論理演算を行い、4ビット桁上げ信号 C^{4bit} を生成する回路である。信号 $I_{1,2}$ と信号 $H_{2,3}$ の論理積をとり、この論理積と信号 $H_{0,1}$ の論理和をとり、この論理和と伝搬信号 p_0 の論理積をとることにより、桁上げ信号 C^{4bit}

6

を生成する。この回路は、プリチャージ用PMOSトランジスタ131と、信号 p_0 , $H_{0,1}$, $H_{2,3}$, $I_{1,2}$ が入力されるNMOSトランジスタ132, 133, 134, 135と、インバータ136から構成されている。

【0025】エバリュエーション期間において、図4の回路により桁上げ信号 C^{4bit} が生成される時間は、縦に3個並んだNMOSトランジスタ132, 133, 134がダイナミックノードをディスチャージする時間と、インバータ136が動作する時間との和に依存している。

【0026】ここまで図1～図4の回路動作を個別に述べてきたが、データ a_i と b_i から4ビットの桁上げ信号 C^{4bit} が生成されるまでの信号処理時間は、前段の図1～図3の回路動作時間と後段の図4の回路動作時間との合計になる。前段の図1～図3の回路において動作速度を比較すると、上述したように、図1の回路は縦積みNMOSトランジスタ数が1個であり、図2と図3の回路は縦積みNMOSトランジスタ数が2個であるので、図2または図3の回路の方が図1の回路より遅く動作する。したがって、第1の実施例の総合的な桁上げ信号生成時間は、図2または図3の回路の一方と図4の回路の動作時間によって決まる。すなわち、桁上げ信号 C^{4bit} の生成時間は、縦積みNMOSトランジスタ2+3個の動作時間とインバータ2個の動作時間の和になる。

【0027】本発明の第1の実施例と従来例の回路構成を比較すると、従来例では前段の図9と図10の回路で生成信号 g_i と伝搬信号 p_i を生成し、後段の図11の回路で信号 g_i と p_i から桁上げ信号 C^{4bit} を生成している。これに対して、本実施例では前段の図2と図3の回路で予め生成信号 g_i と伝搬信号 p_i の論理積や論理和を演算しておき、後段の図4の回路で桁上げ信号 C^{4bit} を生成している。言い換えれば、従来例の後段の図11の論理演算の一部を、本実施例では前段の図2と図3の回路で前倒ししている。これによって、第1実施例は従来例に比べて後段回路の演算処理を軽減し、後段回路の縦積みNMOSトランジスタ数を従来の4個から3個に削減することができた。

【0028】本実施例の図2や図3の前段回路は従来例の図9の前段回路に比べて複雑になるが、縦積みNMOSトランジスタ数が同じ2個に保たれるので、図9の回路動作時間と図2や図3の回路動作時間とに顕著な差はない。したがって、前段と後段を合わせた総合的な桁上げ信号生成時間を比較すると、本実施例は従来例に比べて約15%短縮することができ、加算器の高速化を実現することができる。

【0029】なお、本発明による縦積みトランジスタ数の削減すなわち論理演算処理の軽減は、回路方式を問わず本質的に論理回路の高速化に有効である。第1実施例では高速CMOS回路として一般的なCMOSドミノ回路を採用したが、この他にもスタティックCMOS回

10

20

30

40

50

7

路、Bi CMOS 回路、バイポーラECL 回路においても効果があることは明らかである。

【0030】また、本実施例では4ビットの桁上げ信号生成回路を例に挙げたが、本発明をより多ビットの桁上げ信号生成回路に応用できることは言うまでもない。nビット(nは4以上の整数)の桁上げ信号生成回路に拡張する場合には、式(5)と式(6)に倣って2進デー

$$H_{i,i+1} = a_i \cdot b_i + a_{i+1} \cdot b_{i+1} \quad (i = 0, 2, \dots, n-2) \quad \dots(12)$$

$$I_{k,k+1} = (a_k + b_k) \cdot (a_{k+1} + b_{k+1}) \quad (k = 1, 3, \dots, n-3)$$

$$\dots(13)$$

次に、本発明による4ビット桁上げ信号生成回路の第2の実施例を図5～図8により説明する。図5～図8は、それぞれ式(8)～式(11)に対応する論理演算を行うドミノ回路を示す。第2実施例では、第1実施例のように4ビットの2進データ $a_0 a_1 a_2 a_3$ と $b_0 b_1 b_2 b_3$ の論理演算をそのまま行うのではなく、データの反転信号 $a_0 B a_1 B a_2 B a_3 B$ と $b_0 B b_1 B b_2 B b_3 B$ の論理演算を行う。

【0032】図5の回路は $a_0 B$ と $b_0 B$ を受けて伝搬信号 p_0 の反転信号 $p_0 B$ (図1の出力の反転信号)を生成する回路、図6の回路は $a_0 B$ 、 $b_0 B$ 、 $a_2 B$ 、 $b_2 B$ を受けて信号 $H_{1,i+1}$ の反転信号 $H_{1,i+1} B$ (図2の出力の反転信号)を生成する回路、図7の回路は $a_1 B$ 、 $b_1 B$ 、 $a_2 B$ 、 $b_2 B$ を受けて信号 $I_{1,2}$ の反転信号 $I_{1,2} B$ (図3の出力の反転信号)を生成する回路、図8は信号 $p_0 B$ 、 $H_{1,i+1} B$ 、 $I_{1,2} B$ を受けて桁上げ信号 C^{4bit} の反転信号 $C^{4bit} B$ (図4の出力の反転信号)を生成する回路である。

【0033】図5から図8に示した回路は、それぞれ、プリチャージ用PMOSTランジスタ201、211、221、231と、反転信号 $a_0 B$ 、 $b_0 B$ 、 $a_1 B$ 、 $b_1 B$ 、 $a_2 B$ 、 $b_2 B$ 、 $p_0 B$ 、 $H_{0,1} B$ 、 $H_{2,3} B$ 、 $I_{1,2} B$ が入力されるNMOSTランジスタ202、203、212～215、222～225、232～235と、インバータ204、216、226、236から構成されている。

【0034】図5の回路動作は、図面から明らかなように、入力信号と出力信号が異なる以外は前述した図9の回路動作と同じである。同様に、図6は図3と、図7は図2と同じ回路動作を行う。

【0035】図8の回路では、図5からの信号 $p_0 B$ と図6からの信号 $H_{0,1} B$ 、 $H_{2,3} B$ と図7からの信号 $I_{1,2} B$ が入力され、式(11)に対応して、信号 $H_{2,3} B$ と信号 $I_{1,2} B$ の論理和をとり、この論理和と信号 $H_{0,1} B$ の論理積をとり、この論理積と信号 $p_0 B$ の論理和をとることにより、桁上げ信号の反転信号 $C^{4bit} B$ を生成する。エバリュエーション期間において、図8の回路動作時間は、NMOSTランジスタ234と235の少なくとも一方とNMOSTランジスタ232とがダイナミックノードをディスチャージする時間と、インバータ2

8

*タ $a_0 a_1 a_2 a_3 \dots a_n$ と $b_0 b_1 b_2 b_3 \dots b_n$ から式(12)に示す信号 $H_{i,i+1}$ と式(13)に示す信号 $I_{k,k+1}$ を生成し、これらの信号を用いて式(7)に準ずる論理演算を行い、桁上げ信号 C^{nbit} を生成すればよい。

【0031】

36が切り替わる時間との和で決まる。すなわち、図8の回路動作時間の支配要因である縦積みNMOSTランジスタ数は2個である。

【0036】本実施例において、データの反転信号 $a_i B$ と $b_i B$ から4ビットの桁上げ信号の反転信号 $C^{4bit} B$ が生成されるまでの総合的な信号処理時間は、前段の図5～図7の回路動作時間と後段の図8の回路動作時間との合計になる。前段回路の動作時間は図6または図7の2個の縦積みトランジスタとインバータ1個で決まり、後段回路の動作時間は図8の2個の縦積みトランジスタとインバータ1個で決まっているので、信号 $C^{4bit} B$ の総生成時間は、縦積みNMOSTランジスタ2+2個の動作時間とインバータ2個の動作時間の和になる。したがって本実施例の縦積みトランジスタ数は、従来例に比べて2個、第1実施例に比べて1個削減されている。

【0037】本実施例では、前記第1の実施例の図4の正論理を反転論理で組み替えることによって、縦に直列接続されたNMOSTランジスタを横に並列に並び替えて、縦積みトランジスタ数を削減することができた。本実施例はデータ $a_0 a_1 a_2 a_3$ と $b_0 b_1 b_2 b_3$ の反転信号 $a_0 B a_1 B a_2 B a_3 B$ と $b_0 B b_1 B b_2 B b_3 B$ を必要とするが、これらのデータ反転信号が加算器の桁上げ信号生成回路以外の部分で予め生成されている場合には、図5～図7に示した前段回路の直前にデータを反転するための回路を新たに付与する必要はない。また、図8に示した後段回路の出力 $C^{4bit} B$ を受ける論理回路を反転論理で組めば、 $C^{4bit} B$ を反転させて C^{4bit} に戻すための回路を別個に設ける必要がなくなる。したがって、第2実施例の桁上げ信号生成回路の総合的な信号処理時間は、図5から図8の回路動作時間を考慮しておけばよい。結果的に、本実施例の処理時間は従来例に比べて約25%、第1の実施例に比べて約14%短縮されており、第1の実施例に対してさらに加算器の高速化を進めることができる。

【0038】最後に、本発明による16ビット桁上げ信号生成回路の第3の実施例を図12により説明する。本実施例では、第1の実施例で示した4ビット桁上げ信号生成回路をベースに16ビットの桁上げ信号生成回路を構成している。加算器の桁上げ信号生成以外の部分の論

理構成とのバランスを考慮して、4ビット毎に桁上げ信号を出力させている。

【0039】本実施例では、式(2)と第1の実施例の*

*説明の末尾に述べた式(12)、(13)と下記の式(14)～(20)とに対応する論理演算を行う。

【0040】

$$C^{4bit}_{i,i+3} = p_i \cdot (H_{i,i+1} + I_{i+1,i+2} \cdot H_{i+2,i+3}) \quad (i = 0, 4, 8, 12) \quad \dots(14)$$

$$P^{4bit}_{j,j+3} = p_j \cdot I_{j+1,j+2} \cdot p_{j+3} \quad (j = 0, 4, 8) \quad \dots(15)$$

$$H^{8bit}_{k,k+4} = C^{4bit}_{k,k+3} + C^{4bit}_{k+4,k+7} \quad (k = 0, 8) \quad \dots(16)$$

$$I^{8bit}_{4,8} = P^{4bit}_{4,7} \cdot P^{4bit}_{8,11} \quad \dots(17)$$

$$C^{16bit}_{0,15} = P^{4bit}_{0,3} \cdot (H^{8bit}_{0,4} + I^{4bit}_{4,8} \cdot H^{8bit}_{8,12}) \quad \dots(18)$$

$$C^{12bit}_{4,15} = C^{4bit}_{4,7} + I^{8bit}_{4,8} \cdot H^{8bit}_{8,12} \quad \dots(19)$$

$$C^{8bit}_{8,15} = P^{4bit}_{8,11} \cdot H^{8bit}_{8,12} \quad \dots(20)$$

図12に示す16ビット桁上げ信号生成回路は、大きく分けて、4ビット毎の桁上げ信号の生成を行う論理ブロック1101～1104と、これらの論理ブロックから出力された信号をまとめて16ビットの桁上げ信号を生成する論理ブロック1201から成る。

【0041】論理ブロック1101には、16ビットのうち0ビット目から3ビット目のデータ $a_0 a_1 a_2 a_3$ と $b_0 b_1 b_2 b_3$ が入力される。論理ブロック1101に含まれる回路1001と1002は式(12)の演算を行うドミノ回路(図2と同様)、回路1003と1005は式(2)の演算を行う回路(図1と同様)、回路1004は式(13)の演算を行う回路(図3と同様)である。回路1011は、回路1001の出力 $H_{0,1}$ と回路1002の出力 $H_{2,3}$ と回路1003の出力 p_0 と回路1004の出力 $I_{1,2}$ とを受けて式(14)の演算を行い、0～3ビット目の4ビット分の桁上げ信号 $C^{4bit}_{0,3}$ を出力する回路である(図4と同様)。回路1012は、回路1003の出力 p_0 と回路1004の出力 $I_{1,2}$ と回路1005の出力 p_3 とを受けて式(15)の演算を行い、信号 $P^{4bit}_{0,3}$ を出力する回路である(回路図は示していないが当該技術者であればドミノ回路で容易に構成できる)。

【0042】論理ブロック1102、1103の回路構成は論理ブロック1101と同様である。論理ブロック1102は4～7ビット目のデータ $a_4 a_5 a_6 a_7$ と $b_4 b_5 b_6 b_7$ を受けて、4ビット分の桁上げ信号 $C^{4bit}_{4,7}$ と信号 $P^{4bit}_{4,7}$ を出力する。論理ブロック1103は8～11ビット目のデータ $a_8 a_9 a_{10} a_{11}$ と $b_8 b_9 b_{10} b_{11}$ を受けて、桁上げ信号 $C^{4bit}_{8,11}$ と信号 $P^{4bit}_{8,11}$ を出力する。論理ブロック1104の回路構成は、論理ブロック1101から回路1005と1012を除いたものと同様である。論理ブロック1104は12～15ビット目のデータ $a_{12} a_{13} a_{14} a_{15}$ と $b_{12} b_{13} b_{14} b_{15}$ を受けて、桁上げ信号 $C^{4bit}_{12,15}$ を出力する。

【0043】論理ブロック1201は回路1021～1023と回路1031～1033から成る。回路1021は、論理ブロック1101と1102から出力された

信号 $C^{4bit}_{0,3}$ と $C^{4bit}_{4,7}$ を受けて式(16)に示す論理和演算を行い信号 $H^{8bit}_{0,4}$ を出力する。回路1023も、回路1021と同様に信号 $C^{4bit}_{8,11}$ と $C^{4bit}_{12,15}$ を受けて信号 $H^{8bit}_{8,12}$ を出力する。回路1022は、論理ブロック1102と1103から出力された信号 $P^{4bit}_{4,7}$ と $P^{4bit}_{8,11}$ を受けて式(17)の演算を行い、信号 $I^{8bit}_{4,8}$ を出力する。

【0044】回路1031は、回路1012から出力された信号 $P^{4bit}_{0,3}$ と回路1021～1023から出力された信号 $H^{8bit}_{0,4}$ 、 $H^{8bit}_{8,12}$ 、 $I^{8bit}_{4,8}$ を受けて16ビット全体の桁上げ信号 $C^{16bit}_{0,15}$ を生成する回路であり、回路図は図4と同様である。回路1032は、回路1012から出力された信号 $C^{4bit}_{4,7}$ と回路1022、1023から出力された信号 $H^{8bit}_{8,12}$ 、 $I^{8bit}_{4,8}$ を受けて4～15ビット目の12ビット分の桁上げ信号 $C^{12bit}_{4,15}$ を生成する回路である。回路1033は、回路1013から出力された信号 $P^{4bit}_{8,11}$ と回路1023から出力された信号 $H^{8bit}_{8,12}$ を受けて8～15ビット目の8ビット分の桁上げ信号 $C^{8bit}_{8,15}$ を生成する回路である。

【0045】以上のようにして、本実施例では、論理ブロック1201から4ビット毎に区別した桁上げ信号 $C^{16bit}_{0,15}$ と $C^{12bit}_{4,15}$ と $C^{8bit}_{8,15}$ と $C^{4bit}_{12,15}$ を出力させている。

【0046】上記第3の実施例によれば、各論理ブロック1101～1104において第1実施例で述べたと同様の高速化が図られている。各要素回路で縦積みNMOストランジスタ数が3個以下に抑えられており、ドミノ回路の高速動作が可能である。第3の実施例は従来技術による回路構成に比べて、16ビットのデータから桁上げ信号 $C^{16bit}_{0,15}$ が生成されるまでの総合信号処理時間を約15%短縮することができた。

【0047】なお、上記第3の実施例は第1の実施例の正論理回路に基づいて構成したが、第2の実施例に示したような反転信号による負論理回路でも構成できることは明らかである。負論理回路で構成した場合、第3の実施例は従来技術による回路構成に比べて、16ビットのデータから桁上げ信号 $C^{16bit}_{0,15}$ が生成されるまでの

総合信号処理時間を約23%短縮することができる。また、第3の実施例では16ビット桁上げ信号生成回路の一例を示したが、本発明による論理回路構成を32ビットや64ビットに拡張できることは言うまでもない。本発明に基づく論理演算の先行処理とこれによる縦積みトランジスタ数の削減は、さまざまな桁上げ信号生成回路の高速化にとって広く効果を発揮し得る。

【0048】

【発明の効果】本発明によれば、桁上げ信号生成回路を構成する前段回路と後段回路に関して、前段回路で後段回路の論理演算の一部を先行処理することにより、前段回路の演算を担う縦積みNMOSTランジスタ数を増加させずに後段回路の縦積みトランジスタ数を削減できる。したがって、前後段合わせた総合的な桁上げ信号生成時間を短縮でき、情報処理装置における加算器を高速化できる効果がある。

【0049】また、本発明によれば、桁上げ信号生成回路を負論理回路（反転信号による論理回路）で構成することにより、後段回路の縦積みのNMOSTランジスタを横に並び替えることができる。したがって、後段回路の演算時間の支配要因である縦積みトランジスタ数を削減でき、さらに桁上げ信号生成回路については加算器を高速化できる効果がある。

【0050】なお、本発明による縦積みトランジスタ数の削減すなわち論理演算処理の軽減は、回路方式に関わらず本質的に論理回路の高速化に有効であり、多ビットの桁上げ信号生成回路においてその効果を大に発揮する。

【図面の簡単な説明】

【図1】本発明の第1実施例において伝搬信号 p_0 を生成する回路の回路図。

【図2】本発明の第1実施例において信号 $H_{i,i+1}$ を生成する回路の回路図。

【図3】本発明の第1実施例において信号 $I_{1,2}$ を生成する回路の回路図。

【図4】本発明の第1実施例において4ビット桁上げ信号を生成する回路の回路図。

【図5】本発明の第2実施例において伝搬信号 p_0 の反転信号 P_0B を生成する回路の回路図。

【図6】本発明の第2実施例において信号 $H_{i,i+1}B$ を生成する回路の回路図。

【図7】本発明の第2実施例において信号 $I_{1,2}B$ を生成する回路の回路図。

【図8】本発明の第2実施例において4ビット桁上げ信号反転信号を生成する回路の回路図。

【図9】従来技術において生成信号 g_i を生成する回路の回路図。

【図10】従来技術において伝搬信号 p_i を生成する回路の回路図。

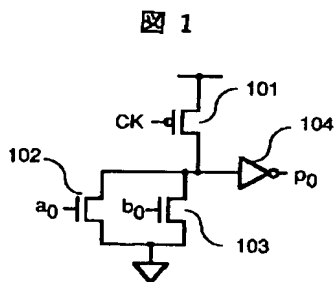
【図11】従来技術において4ビットの桁上げ信号を生成する回路の回路図。

【図12】本発明による第3実施例の16ビット桁上げ信号生成回路の論理図。

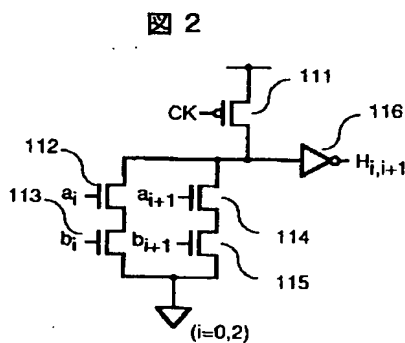
【符号の説明】

101, 111, 121, 131, 201, 211, 221, 231, 301, 311, 321…プリチャージ用PMOSTランジスタ、102, 103, 112, 113, 122, 123, 132, 133, 202, 203, 212, 213, 222, 223, 232, 233, 302, 303, 312, 313, 322, 323…NMOSTランジスタ、104, 116, 126, 136, 204, 216, 226, 236, 304, 314, 329…CMOSインバータ、1001~1005, 1011, 1012, 1021~1023, 1031~1033…回路、1101~1104, 1201…論理ブロック。

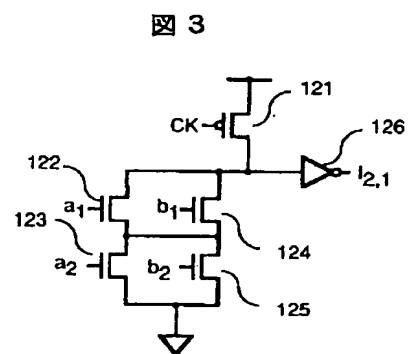
【図1】



【図2】

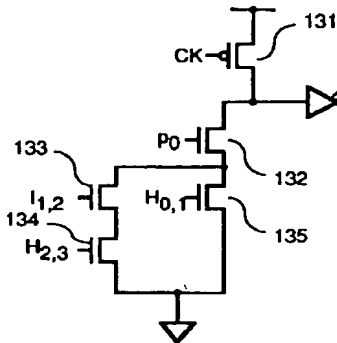


【図3】



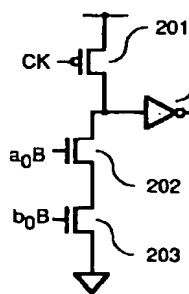
【 図4 】

図 4



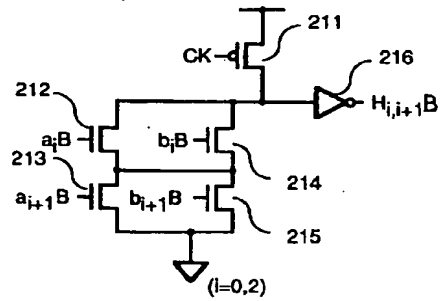
【 図5 】

図 5



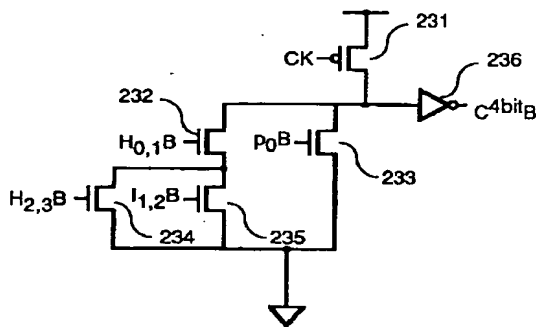
【 図6 】

図 6



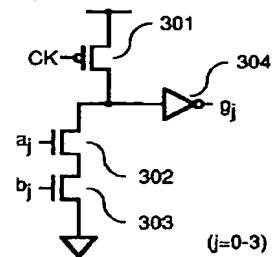
【 図8 】

図 8



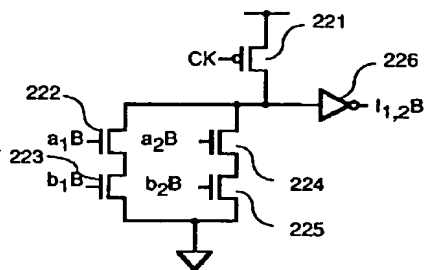
【 図9 】

図 9



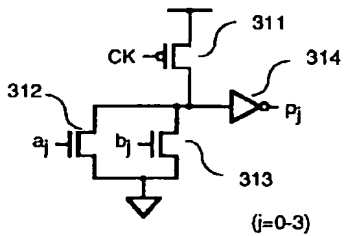
【 図7 】

図 7



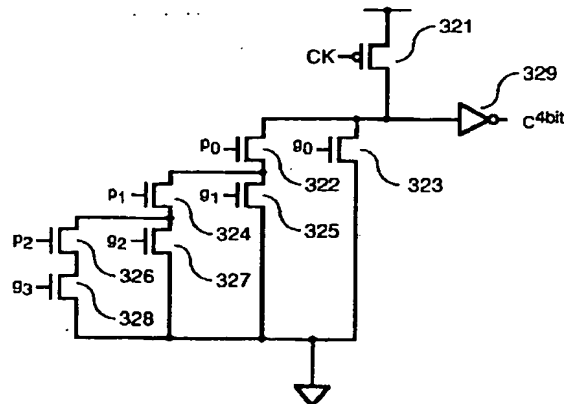
【 図10 】

図 10



【 図11 】

図 11



【 図12 】

図 12

